

Computación Avanzada

Profesores: Ing. César Aranda
Ing. Rubén Ruiz
Ciclo: 2008



Open Minds



Help Desk

**Horacio Branciforte
Enzo Cappa
Agustín Collado**

Índice de contenido

Objetivos.....	4
Descripción.....	4
Listados.....	7
Vista de Casos de Uso.....	8
Diagrama de Casos de Usos.....	8
Definición de Casos de Uso.....	9
1. Registrar.....	9
2. Realizar consulta.....	12
3. Asistir en línea.....	16
4. Gestionar consultas.....	20
5. Administrar usuarios.....	23
6. Verificar Identidad.....	26
7. Consultar historial y estadísticas	28
8. Ver Consultas.....	29
9. Consultar Información.....	32
10. Agregar Información.....	34
Árbol de interfaces.....	36
Vista de diseño.....	37
Diagrama de clases.....	37
Diagrama de clases de la B.D.....	37
Diccionario de clases.....	38
Listado de clases de Modelo.....	38
Listado de clases de Controlador.....	39
Detalle Modelo.....	39
Detalle Controlador.....	41

Objetivos

Fundamentales:

El objetivo del proyecto es realizar un sistema multiusuario de ayuda al cliente (help desk) para dar soporte al paquete de software de oficina OpenOffice.

Secundarios:

- Mostrar estadísticas sobre el trabajo operadores
- Brindar una interfaz amigable tanto a los usuarios como a los operadores
- Generar información historial que sirva como base de conocimientos

Descripción

Alcance

Se brinda una sitio web entre el usuario, los operadores y las personas encargadas de coordinarlos.

El sistema contemplara tres funcionalidades principales:

1- Realizar consultas por parte de los usuarios: mediante un sistema en linea, a través de un sistema de chat o de manera diferida a través de consultas almacenadas que luego serán presentadas al staff de OpenMinds.

2- Respuesta por parte de los operadores: Tendrán la posibilidad de aceptar conversaciones con usuarios para responder consultas en linea, también se presentaran las consultas diferidas y la opción para poder responderlas.

3- Presentación de estadísticas para los Auditores: Podrán ver las estadísticas de uso del sistema, con filtros que permitan agrupar la información y también gráficos que permitan un mejor análisis.

La alta de usuarios se realizara por un sistema de registro, los mismos la solicitaran. También podrán modificar sus perfiles.

Para mejorar la atención al cliente se permite clasificar las consultas, por temas y por tipo de consulta. Entre los datos enviados se permite el envío de archivos imágenes que puedan ayudar la resolución del problema.

Para la gestión y monitoreo del sistema se le brinda a un usuario Auditor la posibilidad de poder promocionar privilegios, eliminar, banear a usuarios, operadores y auditores.

El proyecto sera desarrollado utilizando Javascript desde el lado del cliente, ruby desde el lado del servidor acompañado por el framework Rails, utilizando patrones MVC en un diseño en capas.

Limites

El sistema no cuenta con envío de datos encriptados

Diseño gráfico

Se eligió una plantilla con colores y tipografías similares a las utilizadas en OpenOffice. Esta plantilla se utilizará como base para todas las interfaces, utilizando como encabezado en todos los caso el nombre de la organización y logo de OpenOffice.

En el portal se brinda una presentación de la organización, una pequeña descripción de uso del sistema y una cita a las palabras de Richard Stallman acerca del software libre, de manera que se manifieste mejor la filosofía de la organización. Además en el mismo se deben identificar los usuarios (de todos los niveles) para ingresar a la mesa de ayuda y el punto a partir del cual se pueden registrar los nuevos. A continuación se presenta una toma de pantalla del portal:

Open Minds

¿Quiénes somos?

Open Minds es una empresa comprometida con el soporte para aplicaciones libres de Ofimática, entre nuestros servicios el usuario puede encontrar centros de ayuda en lo que respecta a casi todo tipo de software libre, contratando así el soporte propio de cada aplicación, en este caso la gama de OpenOffice 2 de Sun Microsystems.

¿Como puedo realizar una consulta?

Una vez contratado el servicio, debe registrar un usuario y luego nuestros operadores le darán el alta para que pueda realizar consultas.

Una vez dado de alta Usted cuenta con dos tipos de consultas, por un lado la consulta "on Line" que serán contestadas por técnicos de nuestro Staff de profesionales o tiene la consulta "Off line" o diferida, por la cual usted envía su inquietud y esta será respondida en la brevedad.

¿Porque Usar soft libre? (por Richard Stallman)

El «software libre» es una cuestión de libertad, no de precio. Para comprender este concepto, debemos pensar en la acepción de libre como en «libertad de expresión» y no como en «barra libre de cerveza». Con software libre nos referimos a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, modificar y mejorar el software.

Nos referimos especialmente a cuatro clases de libertad para los usuarios de software:

Libertad 0: la libertad para ejecutar el programa sea cual sea nuestro propósito.

Libertad 1: la libertad para estudiar el funcionamiento del programa y adaptarlo a tus necesidades el acceso al código fuente es condición indispensable para esto.

Libertad 2: la libertad para redistribuir copias y ayudar así a tu vecino.

Libertad 3: la libertad para mejorar el programa y luego publicarlo para el bien de toda la comunidad —el acceso al código fuente es condición indispensable para esto.

"El «código» es la tecnología que hace que los ordenadores funcionen. Está inscrito en el software o grabado en el hardware, es el conjunto de instrucciones, primero escritas como palabras, que dirigen la funcionalidad de las máquinas. Estas máquinas (ordenadores) definen y controlan cada vez más nuestras vidas. Determinan cómo se conectan los teléfonos y qué aparece en el televisor. Deciden si el vídeo puede enviarse por banda ancha hasta un ordenador. Controlan la información que un ordenador remite al fabricante. Estas máquinas nos dirigen. El código dirige estas máquinas. ¿Qué control deberíamos tener sobre el código? ¿Qué comprensión? ¿Qué libertad debería haber para neutralizar el control que permite? ¿Qué poder?:

"Software libre» es la respuesta a un mundo construido mediante código.

Nombre de usuario

Password

[Entrar](#)

[Registrarse](#)

Listados

Referencias de funciones del sistema

- ✓ Registro de usuarios
- ✓ Validación de datos de usuarios
- ✓ Modificación de datos de usuarios
- ✓ Clasificación de consultas por aplicación consultada y categoría.
- ✓ Búsquedas y filtrado de consultas para los operadores
- ✓ Tres niveles de privilegios de usuario (usuarios, operadores, auditores)
- ✓ Sistema de Historial (log) de los eventos realizados en el sistema

Descripción de los actores del sistema

Usuario: Persona/cliente habilitado para poder realizar consultas y comunicarse con los operadores.

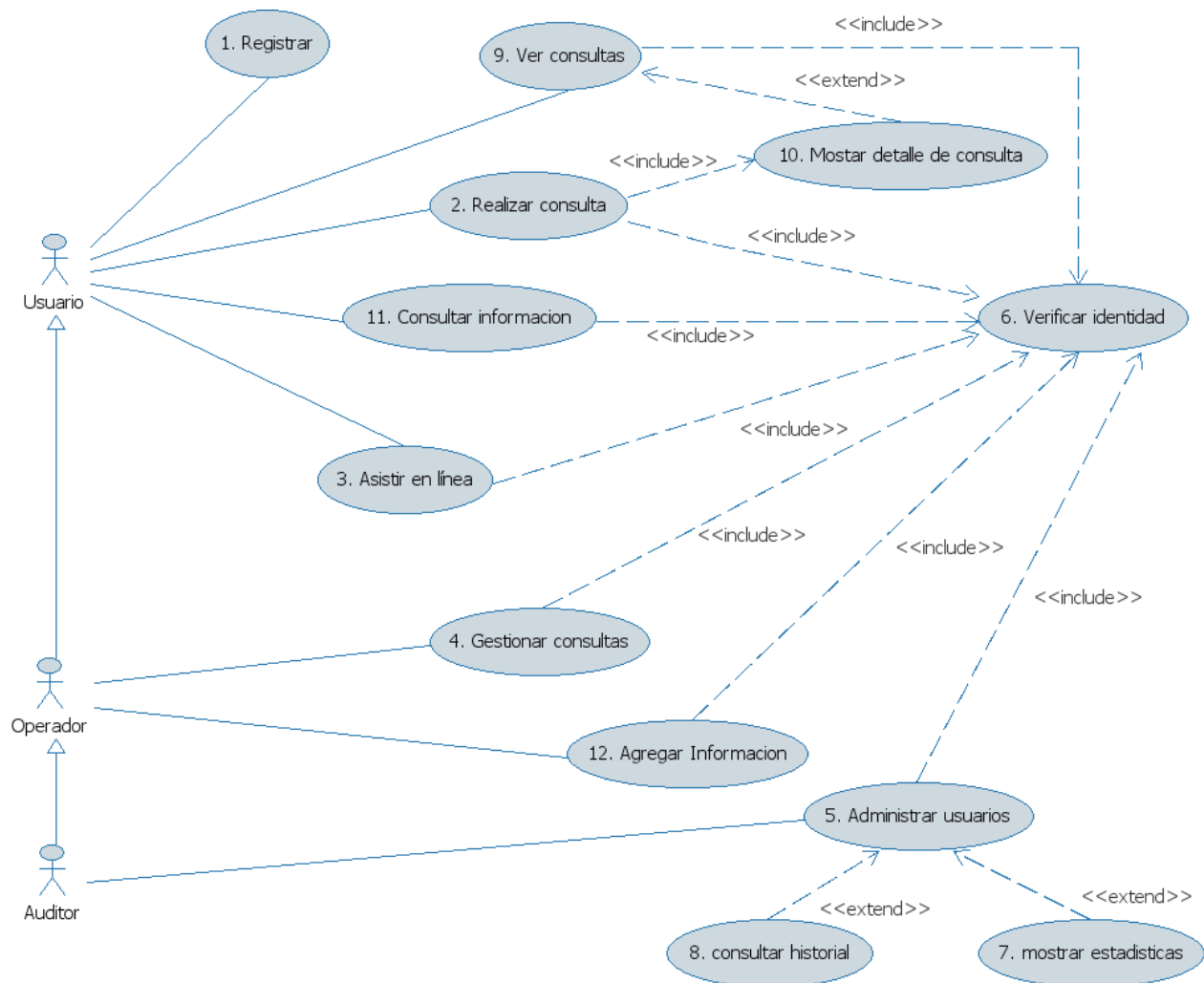
Operador: Usuario con acciones extendidas para poder derivar/responder consultas.

Auditor: Operador con acciones extendidas para poder Administrar usuarios/operadores.

Vista de Casos de Uso

Diagrama de Casos de Usos

Nota: Todos los diagramas cumplen con UML 1.3



Definición de Casos de Uso

1. Registrar

Actores

Usuario: Persona que quiere hacer uso del sistema

Descripción

El usuario ingresa sus datos personales para ser almacenados en el sistema y obtiene un nombre de usuario y contraseña

Precondiciones

El usuario no debe estar registrado en el sistema

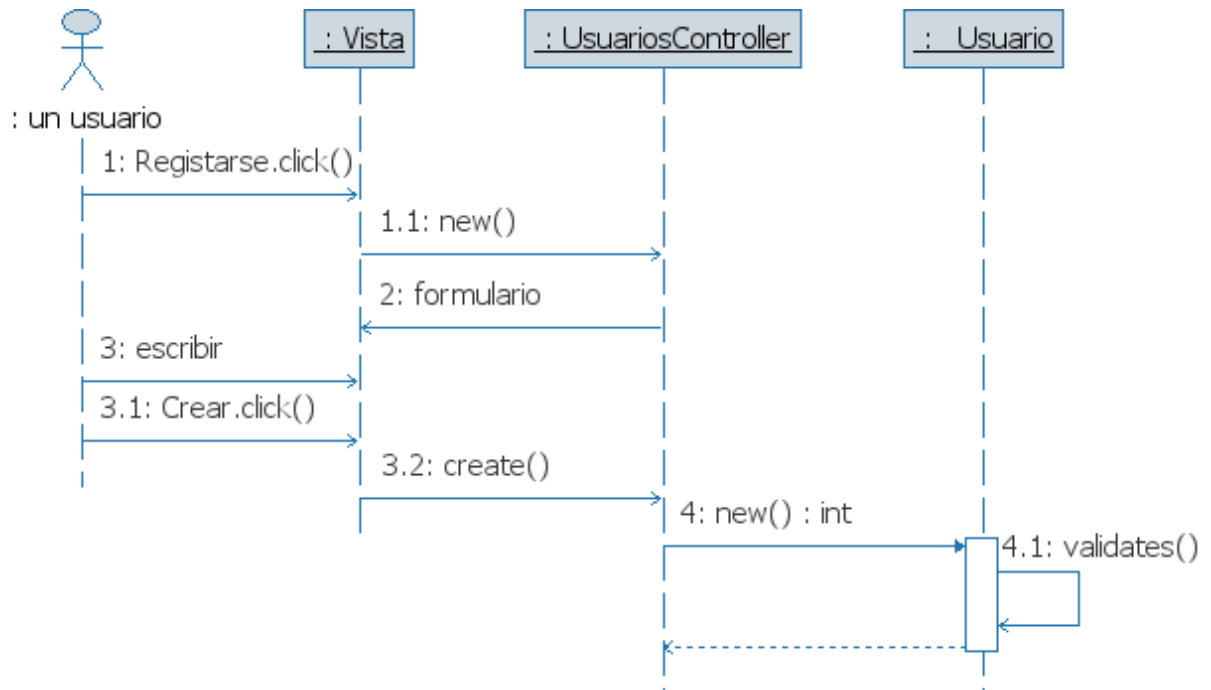
Postcondiciones

Se agrega el usuario al sistema

Curso típico de eventos

<u>Actor</u>	<u>Sistema</u>
1. Presiona el enlace de "Registrarse"	
	2. Muestra el formulario de registro.
3. Escribe sus datos personales y confirma la solicitud presionando el botón "Crear".	
	4. Verifica la validez lógica de los datos ingresados y los almacena. El C.U. Termina.

Diagrama de secuencia



Flujos opcionales

Si los datos ingresados son incorrectos, el sistema informa del error encontrado y solicita que el usuario lo solucione.

Interfaz de usuario

Formulario de creación de usuario

Registro de nuevo usuario

Datos personales

Nombre

Apellido

Email

Fecha de nacimiento

Domicilio:

País

Provincia

Datos de usuario:

Nombre de usuario

Password

[Volver](#)

usuarios/new.rhtml

2. Realizar consulta

Actores

Usuario: Persona que solicita la respuesta a una inquietud

Descripción:

Un usuario realiza una consulta, y el sistema la registra.

Precondiciones

El usuario tiene que estar identificado en el sistema

Postcondiciones

Se agrega una nueva consulta al sistema

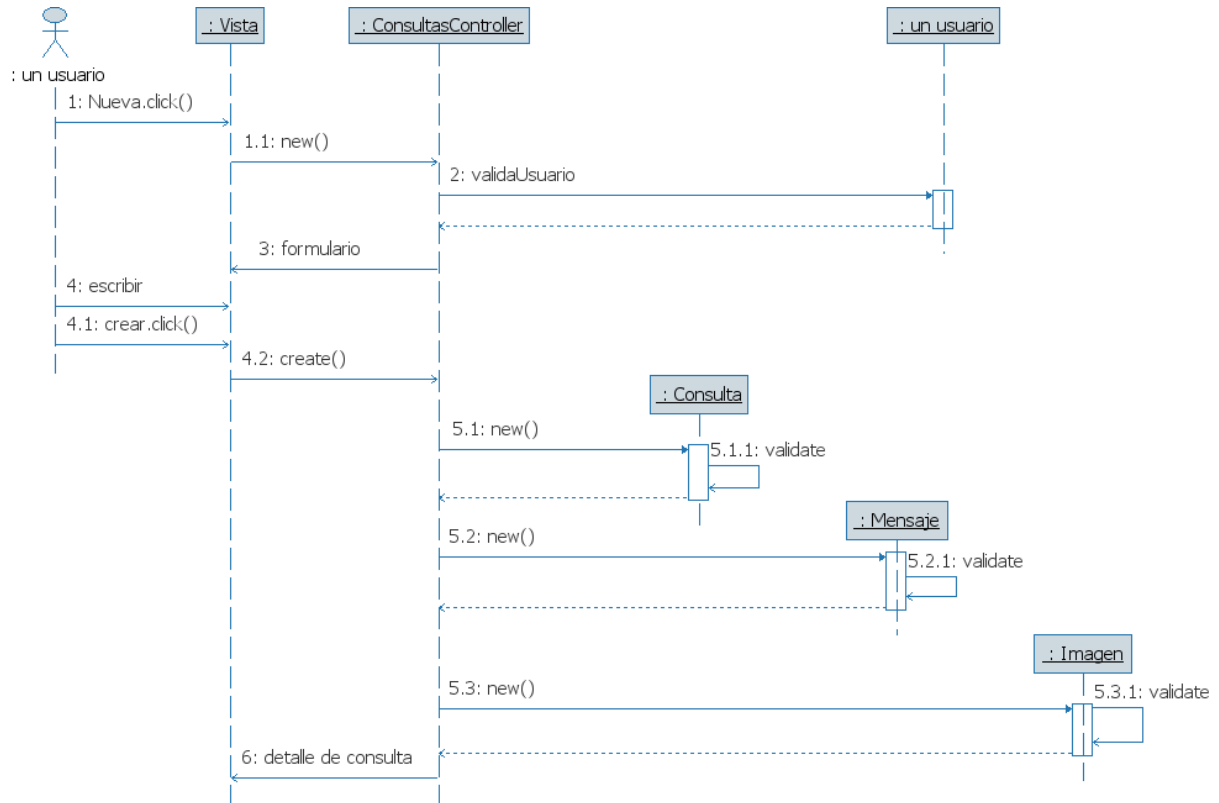
Curso típico de eventos

<u>Actor</u>	<u>Sistema</u>
1. Presiona el enlace "Nueva" en la sección "Mis consultas".	
	2. <<Include>> C.U. Verificar Identidad.
	3. Muestra el formulario de nueva consulta.
4. Escribe su consulta y presiona el botón "Crear".	
	5. Verifica la validez lógica de los datos ingresados y los almacena.
	6. Muestra el detalle de la nueva consulta. El C.U. Termina.

Flujos opcionales

Si los datos ingresados son incorrectos, el sistema informa del error encontrado y solicita que el usuario lo solucione.

Diagrama de secuencia



Interfaces de usuario

Sección "Mis consultas"

Mis consultas

[Nueva](#)

Ordenar

Orden: Fecha  Ordenar

Mostrar


Estado: Pendiente  Mostrar

Fecha	Estado	Titulo	
12/06/08	Pendiente	asdas	Detalle Cerrar
12/06/08	Finalizada	asdas	Detalle

Filtrar

Fecha

Desde: Hasta:

Estado: Pendiente 

Título: Buscar

[Nueva](#)

[consultas/list.rhtml](#)

Nueva consulta

Categoría Aplicación

Título

Descripcion

Adjunto

[Volver](#)

consultas/new.rhtml

3. Asistir en línea

Actores

Usuario: Persona que solicita la respuesta a una inquietud

Operador: Persona capacitada y responsable de responder a la inquietud

Descripción

Un actor responde en directo la consulta del usuario.

Precondiciones

El usuario y el operador están identificados en el sistema

Hay un operador disponible para responder la consulta

Postcondiciones

Se registra la consulta en el sistema

Curso típico de eventos

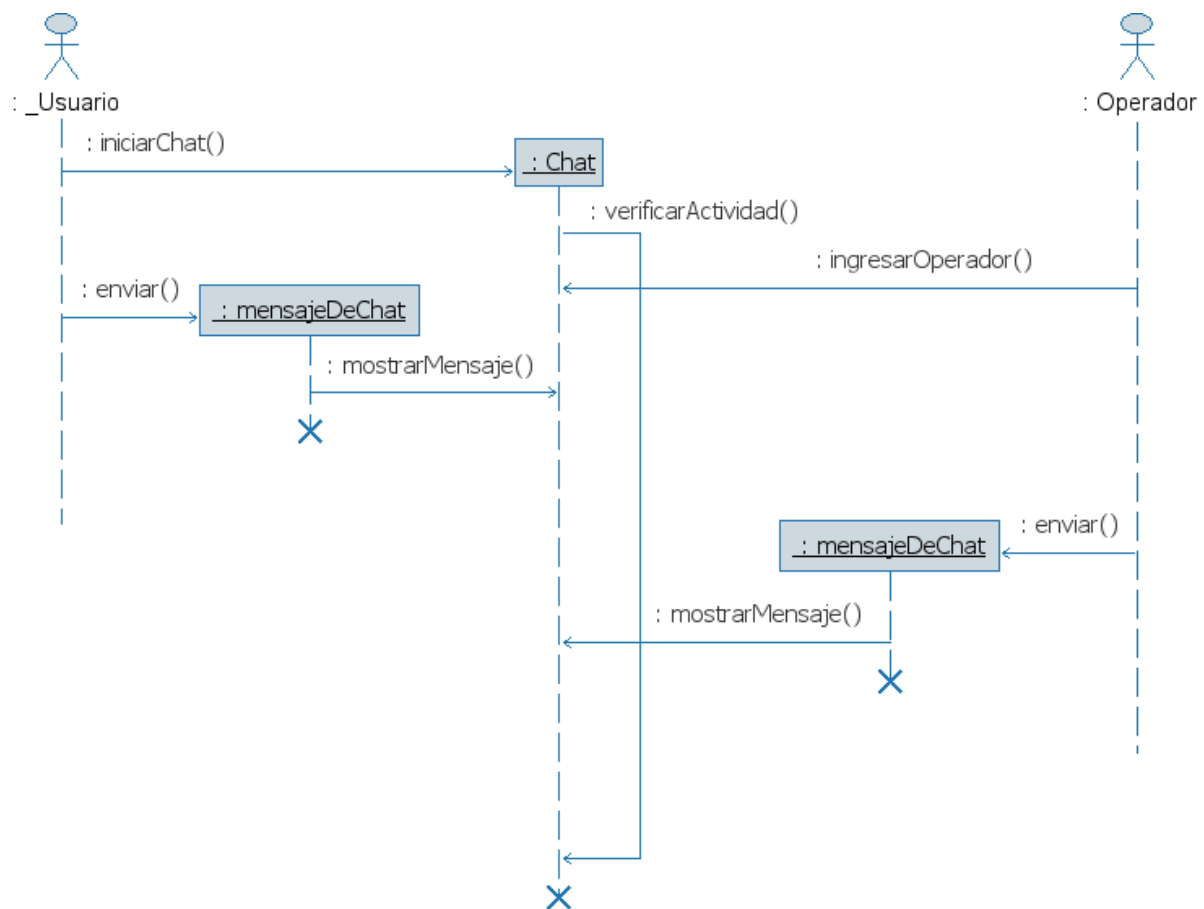
<u>Actor: Usuario</u>	<u>Actor: Operador</u>	<u>Sistema</u>
1. Solicita asistencia en línea presionando el enlace "Asistencia en línea".		
		2. <<Include>> C.U. Verificar Identidad.
		3. Agrega la solicitud a la lista de pendientes y muestra al usuario la ventana de asistencia.
	4. Acepta la solicitud desde la lista.	
5. Escribe la consulta y presiona el botón enviar..		

		6. Almacena la consulta y la muestra en la ventana del operador.
	7. Escribe la respuesta y presiona el botón enviar.	
		8. Almacena la respuesta y la muestra en la ventana del usuario.
		9. Tras 5 minutos de inactividad, el C.U. Termina.

Flujos opcionales:

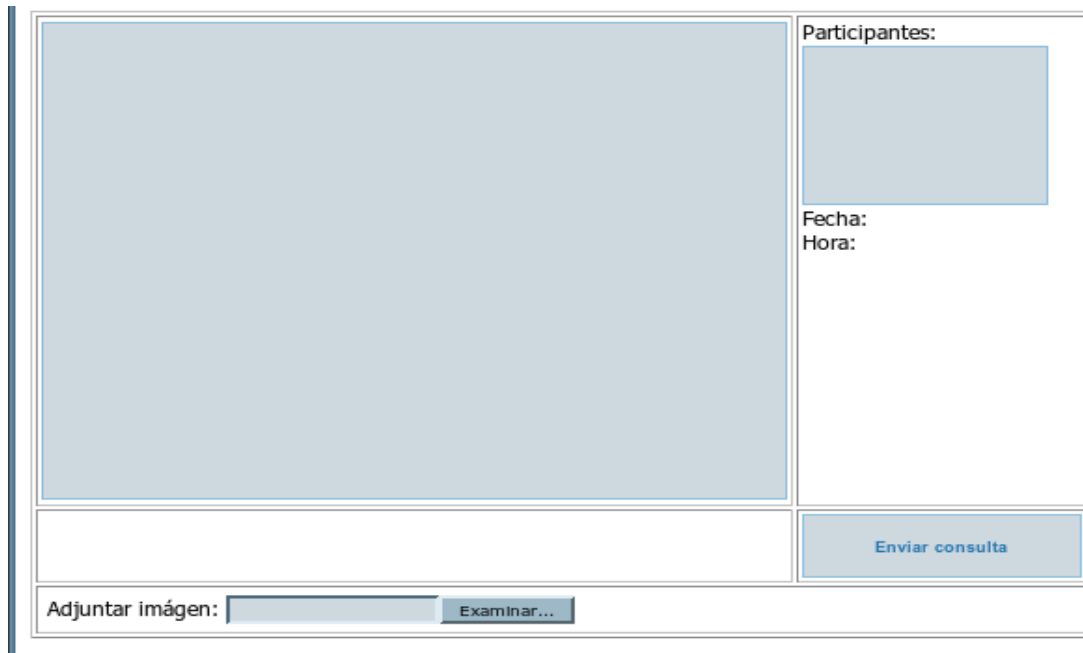
La secuencia entre 5 y 8 se puede repetir indefinidas veces.

Diagrama de secuencia



Interfaces de usuario

Interfaz de asistencia en línea



chat/new.rhtml

Lista de consultas pendientes

Asistencia en línea

Hora	Usuario	
11:00	Pepe	Asistir

chat/list.rhtml

4.Gestionar consultas

Actores

Operador: Persona que está encargada de responder las consultas.

Descripción

El operador procesa las solicitudes pendientes.

Precondiciones

El operador está identificado en el sistema

Postcondiciones

Se modifica el estado de las consultas pendientes

Curso típico de eventos

Actor	Sistema
1. Presiona el enlace "Consultas Pendientes"	
	2. <<include>> C.U. verificar identidad.
	3. Muestra la lista de consultas pendientes
4. Puede responder o cambiar clasificación de una consulta.	
	5. Guarda las acciones realizadas. El CU termina.

Flujos opcionales:

Si los datos ingresados por el operador son incorrectos, el sistema pide que el mismo solucione el problema.

Interfaces de usuario

Lista de consultas pendientes

Consultas pendientes

Mostrar

Categoría:

Soporte Técnico

Mostrar

Aplicación:

Writer

Mostrar

Fecha	Usuario	Asunto	
12/06/08	Admin	asdas	Detalle Cerrar
12/06/08	Admin	Esta es el detalle de un	Detalle Cerrar
12/06/08	Admin	asd	Detalle Cerrar
12/06/08	Admin	asd	Detalle Cerrar
12/06/08	Admin	Ejemplo	Detalle Cerrar

Filtrar

Fecha

Desde:

Hasta:

Título:

Buscar

consultas/listOp.rhtml

Detalle de consulta

Esta es una consulta de u

Tema	Esta es una consulta de usuario
Tengo muchas dudas....	
Mostrar imagen adjunta	
Tema	Esta es la respuesta del operador
Tenes que leer mas	
Mensaje nuevo Atras	

consulta/show.rhtml

Formulario de respuesta

Nuevo mensaje

Asunto	<input type="text"/>
Descripcion	<div></div>
Adjunto	<input type="text"/> <input type="button" value="Examinar..."/>
<input type="button" value="Crear"/>	
Volver	

consulta/new.rhtml

5. Administrar usuarios

Actores

Auditor: Persona encargada del sistema y los usuarios.

Descripción

Se audita las consultas procesadas.

Precondiciones

El administrador debe estar identificado en el sistema

Postcondiciones

Se modifican los datos de uno o más usuarios

Curso típico de eventos

<u>Actor</u>	<u>Sistema</u>
1. Presiona el enlace "Administrar usuarios"	
	2. <<include>> CU Verificar la identidad
	3. Muestra la lista de usuarios
4. Realiza alguna operación sobre el usuario:	
4.1. Presiona "Mostrar" para ver el detalle.	
	5. Muestra los detalles del usuario y el C.U. termina.
4.2. Presiona "Destruir" para dar de baja el usuario.	

	5. Muestra una solicitud de confirmación antes de realizar la operación
6. Acepta o rechaza la confirmación	
	7. De acuerdo a la respuesta, ejecuta la acción correspondiente y el C.U. Termina.
4.3. Presiona "Editar" para cambiar los datos de un usuario.	
	5. Muestra el formulario de edición de datos de un usuario.
6. Realiza los cambios correspondientes y presiona "Editar".	
	7. Verifica la validez lógica de los datos y los almacena. El C.U. Termina.

Flujos opcionales:

Si los datos ingresados son incorrectos, el sistema informa del error encontrado y solicita que el auditor lo solucione.

Interfaces de usuario

Lista de usuarios

Lista de usuarios

Nombre	Apellido	Nomusuario	Privilegio	
Admin	Admin	admin	7	Mostrar Editar Destruir
operador	operandez	operador	4	Mostrar Editar Destruir
usuario	usuariez	usuario	1	Mostrar Editar Destruir

Ordenar

Ordenar por Privilegio  Ordenar

Buscar

Buscar

Usuario  Buscar

usuarios/list.rhtml

Formulario de edición de datos de usuario

Editar perfil

Datos personales

Nombre

Apellido

Email

Fecha de nacimiento

Domicilio:

País

Provincia

Datos de usuario:

Nombre de usuario

Password

Privilegio

[Mostrar](#) | [Volver](#)

usuarios/edit.rhtml

6. Verificar Identidad

Actores:

CU Realizar consultas, CU Gestionar consultas, CU Administrar usuarios, CU Asistir en línea

Usuario: persona que quiere identificarse en el sistema

Descripción:

Verifica que el usuario este identificado en el sistema.

Precondiciones

Ninguna

Postcondiciones

Se identifica al usuario

Curso típico de eventos

<u>Actor</u>	<u>Sistema</u>
1. Solicita la verificación de la identidad de un usuario.	
	2. Verifica la identidad del usuario.
	3. Responde con la identidad del usuario. El C.U. termina.

Flujos opcionales

Si el usuario no se puede identificar, se muestra el formulario para que ingrese su nombre de usuario y contraseña.

Interfaces

Formulario de identificación

A screenshot of a login form titled 'Formulario de identificación'. It features two input fields: 'Nombre de usuario' and 'Password'. Below the password field is a button labeled 'Entrar'. At the bottom, there is a link labeled 'Registrarse'.

7. Consultar historial y estadísticas

Actores

Auditor: persona que controla uso del sistema

Descripción

Se presentan datos acerca del uso del sistema.

Precondiciones

Existen datos almacenados en el sistema

El auditor está identificado en el sistema

Postcondiciones

Ninguna

Curso típico de eventos

<u>Actor</u>	<u>Sistema</u>
--------------	----------------

1. Presiona el enlace "Ver historial"	
	2. <<include>> C.U. Verificar identidad
	3. Muestra la lista de informes posibles.
4. Elige uno de los informes en la lista.	
	5. Muestra el informe seleccionado y el C.U. Termina.

Flujos opcionales

Ninguno

8. Ver Consultas

Actores

Usuario: Persona que realizó una o más consultas en el pasado y desea verlas.

Descripción

Se muestran las consultas realizadas por el usuario

Precondiciones

El usuario está identificado en el sistema

Postcondiciones

Ninguna

Curso típico de eventos

<u>Actor</u>	<u>Sistema</u>
1. Presiona el enlace "Mis consultas".	

	2. <<include>> Verificar identidad
	3. Muestra la lista de consultas realizadas por el usuario.
4. Selecciona una de las consultas	
	5. Muestra el detalle de la consulta y el C.U. Termina.

Flujos opcionales

Ninguno

Interfaces de usuario

Lista de consultas para un usuario

Mis consultas

[Nueva](#)

Ordenar

Orden: Fecha  Ordenar

Mostrar


Estado: Pendiente  Mostrar

Fecha	Estado	Titulo	
12/06/08	Pendiente	asdas	Detalle Cerrar
12/06/08	Finalizada	asdas	Detalle

Filtrar

Fecha

Desde: Hasta:

Estado: Pendiente 

Título: Buscar

[Nueva](#)

consultas/list.rhtml

9. Consultar Información

Actores

Usuario: Persona que accede a la base de conocimientos

Descripción

Se muestra la lista de manuales disponibles, de manera que el usuario puede descargarlos.

Precondiciones

Existen manuales en el sistema

El usuario se ha identificado en el sistema

Postcondiciones

Ninguna

Curso típico de eventos

<u>Actor</u>	<u>Sistema</u>
1. Presiona el enlace "Consultar información"	
	2. <<include>> Verificar identidad
	3. Muestra la lista de información disponible.
4. Selecciona un manual y presiona el enlace "Descargar archivo".	
	5. Envía el archivo y el CU termina.

Flujos opcionales

Ninguno

Interfaces de usuario

Lista de manuales

Información disponible

Nombre	Ejemplo
Es un buen ejemplo de lo que se puede hacer con esto	
Descargar archivo	

infos/list.rhtml

10. Agregar Información

Actores

Auditor: Persona con suficientes privilegios para agregar información.

Descripción

Se agrega información a la base de conocimientos.

Precondiciones

El auditor se identificó correctamente en el sistema

Postcondiciones

Se agrega un nuevo manual al sistema

Curso típico de eventos

<u>Actor</u>	<u>Sistema</u>
1. Presiona el enlace "Consultar información"	
	2. <<include>> Verificar identidad
	3. Muestra la lista de manuales con el enlace "Agregar".
4. Presiona el enlace "Agregar"	
	5. Muestra el formulario para registrar un nuevo manual.
6. Completa el formulario con los datos correspondientes y presiona el botón "Agregar".	
	7. Verifica la validez lógica de los datos y los agrega. El C.U. Termina.

Flujos opcionales

Si los datos ingresados son incorrectos, solicita que el auditor solucione los inconvenientes.

Interfaces de usuario

Formulario para agregar manuales

Agregar recurso

Nombre

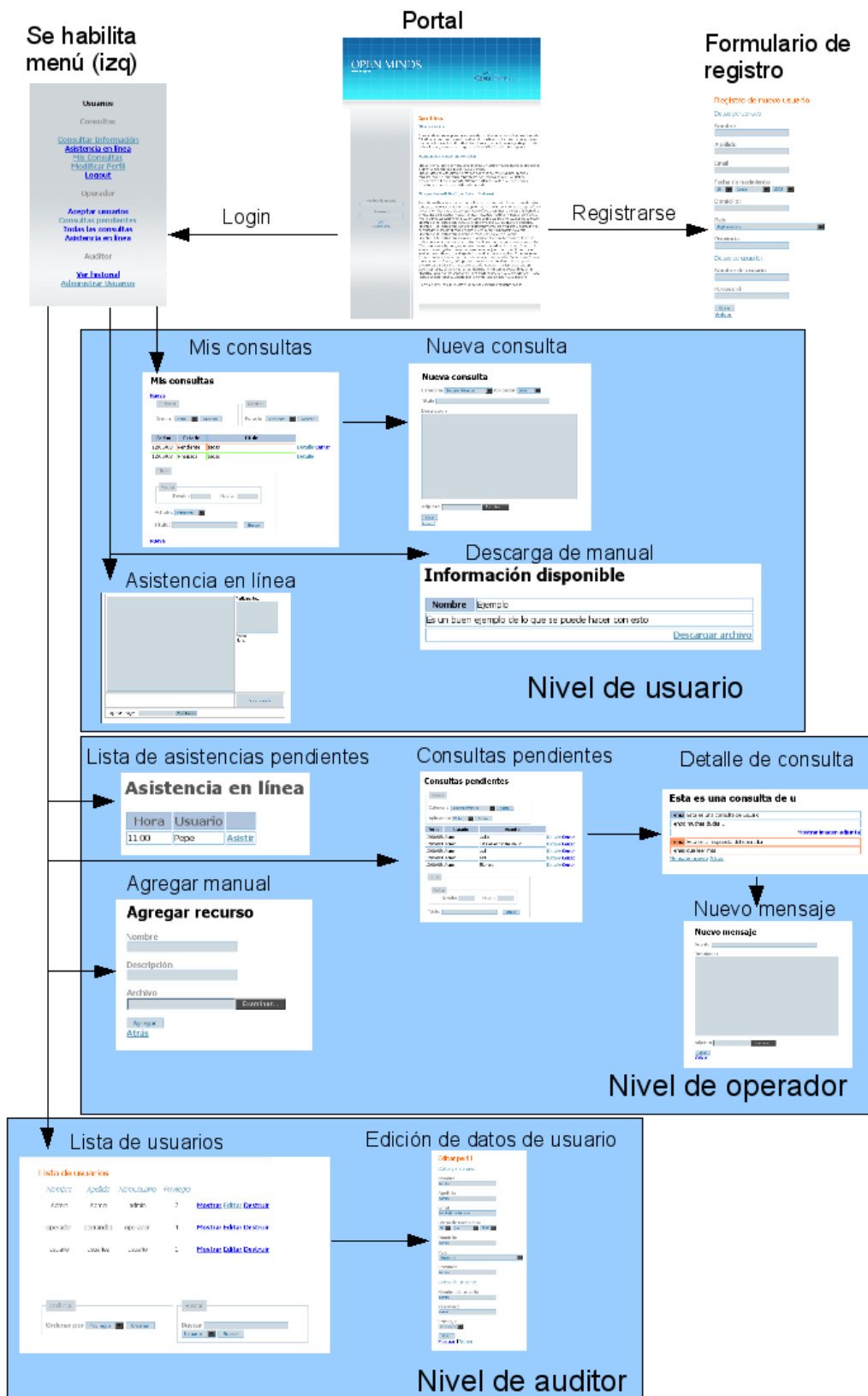
Descripción

Archivo

[Atrás](#)

infos/new.rhtml

Árbol de interfaces



Vista de diseño

Diagrama de clases

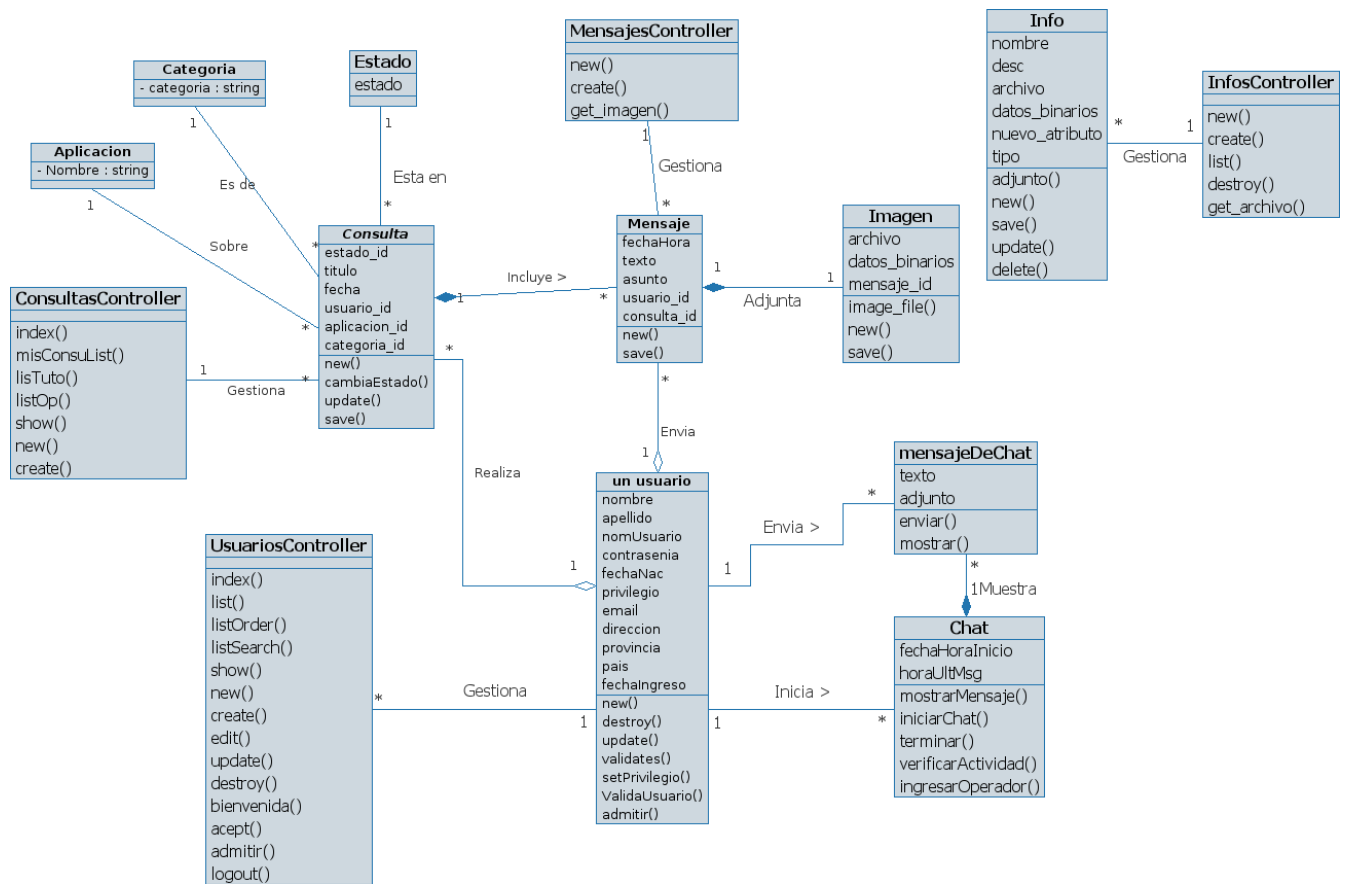
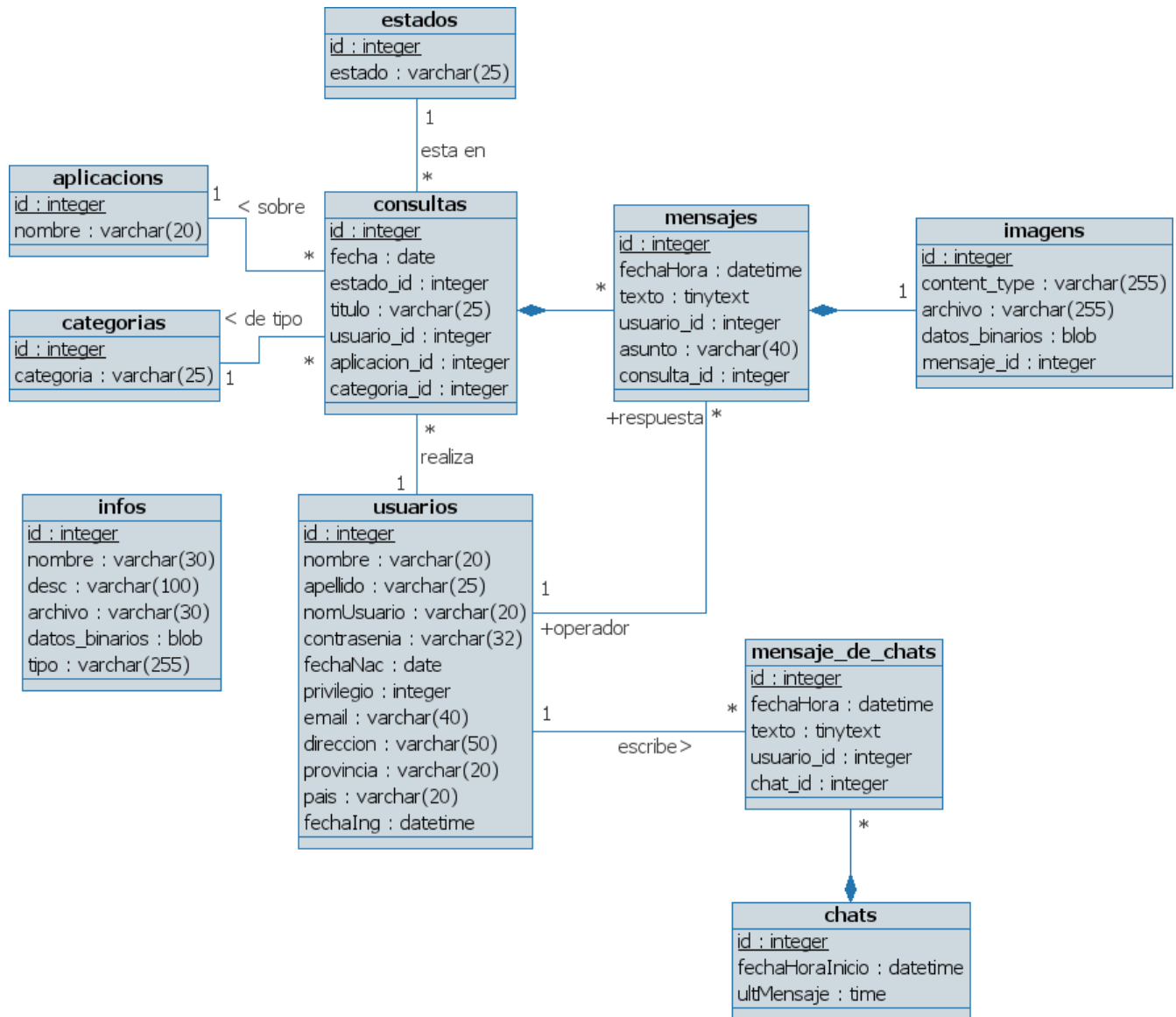


Diagrama de clases de la B.D.



Diccionario de clases

Listado de clases de Modelo

- Usuario
- Consulta
- aplicación
- Categoria
- Estado
- Mensaje
- Imagen
- Mensaje_de_chat
- Chat
- Info

Listado de clases de Controlador

- ConsultasController
- UsuariosController
- MensajesController
- InfosController

Detalle Modelo

Clase: Aplicación

Atributos: Nombre

Relaciones: (1) aplicación tiene (muchas) Consultas

Clase: Categoria

Atributos: categoria

Relaciones: (muchas)consulta es de(1) categoria

Clase: Estado

Atributos: Estado

Relaciones: (1)Estado esta en (muchas) consulta

Clase: Consulta

Atributos: estado_id, titulo, fecha, usuario_id, aplicacion_id, categoria_id

Metodos: new(), cambiaestado(),update(),save()

Relaciones:

- (*)consulta son gestionada por (1) consultaController
- (*)consulta tiene(1) Aplicacion
- (*)consulta tiene(1) Categoria
- (*)consulta tiene(1) Estado
- (1)consulta incluye (*) mensajes
- (*)consulta es realizada por (1) usuario

Clase: Mensaje

Atributos: fechaHora, texto, asunto, usuario_id, consulta_id

Metodos: new(), save()

Relaciones:

- (*) mensaje son gestionados (1) mensajeController
- (*) mensaje son incluidos (1) Consultas
- (1) mensaje puede incluir (1) imagen
- (*) mensaje son enviados por (1) usuario

Clase: Usuario

Atributos: nombre, apellido, nomUsuario, contrasenia, fecha_nac, privilegio, email, direccion, provincia, pais, fechaIngreso.

Metodos: new, destroy,update,validates,setPrivilegios,validaUsuario,adminir

Relaciones:

- (1)usuario envia (*) mensajes
- (1) realiza (*) consultas
- (*) Usuario son gestionados (1) UsuarioController
- (1)Usuario envia (*) MensajesDechat
- (1) Usuario inicia (*) chat

Clase: Chat

Atributos: FechaHoraInicio, fechaHoraInicio, horaUltMsg

Metodos: mostrarMensaje, iniciarChat, terminar, verificarActividad, ingresarOperador

Relaciones:

- (*) Chat son unciados por (1) usuario
- (1) Chat muestra (*) MensajesDeChat

Clase: MensajeDeChat

Atributos: texto, adjunto

Metodos: enviar, mostrar.

Relaciones:

- (*) mensajesDeChat son enviados por (1) Usuario
- (*) MensajeDeChat son mostrados por (1) Chat

Clase: Imagen

Atributos: archivo, datos_binarios, mensaje_id

Metodos: image_file, new, save

Relaciones:

(1)imagen es adjuntada por (1) mensaje

Clase: Info

Atributos: nombre, desc, archivo, datos_binarios, nuevo_atributo, tipo.

Metodos: adjunto, new, save, update, delete,

Relaciones:

(1) Info es gestionado por (1) InfoController

Detalle Controlador

Clase: ConsultaController

Atributos:

Metodos: index, misconsulist, lisTuto, lisOp, show, new, create

Relaciones:

(1)ConsultaController gestiona (*) consultas

Clase: UsuariosController

Atributos:

Metodos: Index, list, listOrder, lisSearch, show, create, edit, update, destroy, bienvenida, accept, adminitr, logout.

Relaciones:

(1)UsuarioController gestiona (*) Usuarios

Clase: MensajeController

Atributos:

Metodos: new, create, get_imagen

Relaciones:

(1)MensajeController gestiona(*) mensajes

Clase: InfosController

Atributos:

Metodos: new, create, list, destroy, get_archivo

Relaciones:

(1) InfoController gestiona (*) Info